

NISHIMURA

Application No. 10/020,437

May 10, 2005

AMENDMENTS TO THE CLAIMS:

This listing of claims will replace all prior versions, and listings, of claims in the application:

1.-11. (canceled)

12. (currently amended) A method for executing an event process in real time on a computer in response to one of a plurality of ~~predetermined~~ events, ~~thesaid~~ method comprising the steps of:

creating a plurality of tasks and assigning one of a plurality of priority levels to each of ~~saidthe~~ the plurality of tasks;

creating a plurality of event processes corresponding to ~~said plurality of predeterminedthe~~ events and assigning one of ~~thesaid plurality of~~ priority levels to each of ~~said plurality ofthe~~ event processes;

dividing ~~said plurality of the~~ event processes into ~~said plurality of the~~ tasks so that each of ~~said plurality ofthe~~ event processes is included in a task of a priority level same as a priority level of the event process;

requesting activation of a certain task which includes an certain event process corresponding to an certain event in response to occurrence of ~~said the~~ certain event;

storing an ID of ~~thesaid~~ certain event process in a queue in response to occurrence of ~~said the~~ certain event ~~if when thesaid certain task in which said~~

NISHIMURA

Application No. 10/020,437

May 10, 2005

~~event process is included~~ includes an event process other than ~~the said~~certain event process;

activating ~~the certain~~said task ~~if~~when a priority level of ~~said~~the certain task is higher than a priority level of an active task;

~~said~~the certain task obtaining the ID of ~~said~~the certain event process from ~~said~~the queue and identifying ~~said~~the certain event process based on the obtained ID ~~if~~when ~~said~~the certain task is activated; and

~~said~~the certain task executing ~~said~~the identified certain event process identified,

wherein at least one specific event is selected from ~~said~~plurality of predeterminedthe events, and an specific event process corresponding to ~~said~~the specific event is included in a dedicated task, which is created at ~~said~~the creating a plurality of tasks step as one of ~~said~~plurality of the tasks for executing only one event process, so that ~~said~~the dedicated task identifies the specific event process ~~corresponding to said specific event~~ for execution without obtaining an ID of the specific event process from ~~said~~the queue when ~~said~~the dedicated task is activated.

13. (new) A processor unit comprising:

a CPU; and

NISHIMURA

Application No. 10/020,437

May 10, 2005

a task scheduling program that causes the CPU to detect an occurrence of an event and to execute an event process corresponding to the event whose occurrence is detected, the task scheduling program including:

a plurality of tasks each of which causes the CPU to execute at least one associated event process,

an activation request program that causes the CPU to detect the occurrence of the event and then to execute a request process that outputs a request for an activation of a first one of the plurality of tasks, and

a real time operating system that causes the CPU to execute an activating process that activates the first one of the plurality of tasks based on one of a plurality of priority levels in response to the request for the activation;

wherein the plurality of tasks include:

a dedicated task that causes the CPU to execute one specific event process corresponding to a specific event that includes a cyclic event which occurs in a cycle in synchronization with time, and

a shared task that causes the CPU to execute at least one of a plurality of non-specific event processes corresponding to a plurality of non-specific events excluding the specific event,

wherein the activation request program stores, in a storage area, identification information by which the CPU identifies the one of the non-specific

event processes based on the request for the activation of the one of the non-specific event processes,

the shared task includes an identification program that causes the CPU to execute an identification process that identifies the one of the non-specific event processes based on the identification information, and

the dedicated task does not include the identification program.

14. (new) A processor unit of Claim 13,

wherein the specific event does not require passing data from the activation request program to the dedicated task.

15. (new) A processor unit of Claim 13,

wherein the shared task is one of a plurality of shared tasks,

each of the shared tasks has been assigned a respective one of the plurality of priority levels, and

a certain non-specific event process is achieved in a shared task whose priority level is equal to a priority level of the certain non-specific event process.

16. (new) A processor unit comprising:

a CPU; and

a task scheduling program that causes the CPU to detect an occurrence of an event and to execute an event process corresponding to the event whose occurrence is detected, for controlling a control object,

wherein the task scheduling program includes:

a plurality of tasks each of which causes the CPU to execute at least one associated event process,

an activation request program that causes the CPU to detect the occurrence of the event and then to execute a request process that outputs a request for an activation of a first one of the plurality of tasks, and

a real time operating system that causes the CPU to execute an activating process that activates the first one of the plurality of tasks based on one of a plurality of priority levels in response to the request for the activation;

wherein the plurality of tasks include:

a dedicated task that causes the CPU to execute one specific event process corresponding to a specific event that includes a cyclic event which occurs in a cycle in synchronization with an operating state of the control object, and

a shared task that causes the CPU to execute at least one of a plurality of non-specific event processes corresponding to a plurality of non-specific events excluding the specific event,

wherein the activation request program stores, in a storage area, identification information by which the CPU identifies the one of the non-specific event processes based on the request for the activation of the one of the non-specific event processes,

the shared task includes an identification program that causes the CPU to execute an identification process that identifies the one of the non-specific event processes based on the identification information, and

the dedicated task does not include the identification program.

17. (new) A processor unit of Claim 16,
wherein the cyclic event occurs in synchronization with a cycle of an engine as the control object.

18. (new) A processor unit of Claim 16,
wherein the specific event does not require passing data from the activation request program to the dedicated task.

19. (new) A processor unit of Claim 16,
wherein the shared task is one of a plurality of shared tasks,
wherein each of the shared tasks has been assigned a respective one of the plurality of priority levels, and

wherein a certain non-specific event process is achieved in a shared task whose priority level is equal to a priority level of the certain non-specific event process.

20. (new) A computer program product of a task scheduling program that causes a CPU to detect an occurrence of an event and to execute an event process corresponding to the event whose occurrence is detected, the computer program product comprising:

a plurality of tasks each of which causes the CPU to execute at least one associated event process,

an activation request program that causes the CPU to detect the occurrence of the event and then to execute a request process that outputs a request for an activation of a first one of the plurality of tasks, and

a real time operating system that causes the CPU to execute an activating process that activates the first one of the plurality of tasks based on one of a plurality of priority levels in response to the request for the activation;

wherein the plurality of tasks include:

a dedicated task that causes the CPU to execute one specific event process corresponding to a specific event that includes a cyclic event which occurs in a cycle in synchronization with time, and

a shared task that causes the CPU to execute at least one of a plurality of non-specific event processes corresponding to a plurality of non-specific events excluding the specific event, wherein the activation request program stores, in a storage area, identification information by which the CPU identifies the one of the non-specific event processes based on the request for the activation of the one of the non-specific event processes,

the shared task includes an identification program that causes the CPU to execute an identification process that identifies the one of the non-specific event processes based on the identification information, and

the dedicated task does not include the identification program.

21. (new) A method comprising:

providing a plurality of tasks including (i) a shared task including a plurality of non-specific event processes corresponding to respective non-specific events, and (ii) a dedicated task including a single specific event process corresponding to a specific event which occurs regularly at predetermined time intervals;

assigning one of a plurality of priority levels to each of said plurality of tasks;

detecting an occurrence of a certain event; and

NISHIMURA

Application No. 10/020,437

May 10, 2005

storing an ID of one of the non-specific event processes in a queue which corresponds to the non-specific event only when the detected certain event is that non-specific event such that an ID of the single specific event process is not stored in the queue when the detected certain event is the specific event.